Amendments to the Specification:

Please replace the paragraph on page 4, entitled, "SUMMARY OF THE INVENTION" with the following amended paragraph:

In order to direct content requests to an appropriate content serving site in a computer network, a phased learning approach is utilized to ensure that, as best as possible, the request is made to content serving site with the shortest delay. In a setup phase, an indirect path return geographic sever load balancer times ~~queries~~ sends transit time requests to all of the individual content serving sites so that the ~~queries~~ transit time requests all arrive at the content serving sites at the same time. Therefore, when the requesting fixed location receives communications from the content serving sites, it can easily tell which content serving site has the least delay by an established race condition. The winner of the race may then be relayed to the indirect path return geographic server load balancer for later usage. In an execution mode, only the *m* fastest content serving sites and *n* other sites (used to test random and new sites) are ~~queried~~ sent a transit time request when a DNS request arrives from the requesting fixed location. The particular *m* fastest content serving sites and *n* other sites may be dynamically updated so as to ensure the most reliable directing of requests. This solution provides a very efficient and effective means by which to determine closest content serving sites while keeping load balancer-created traffic at a minimum.

Please replace the first paragraph on page 8 with the following amended paragraph:

With the knowledge of the response times of each server 56a, 56b, 56c, the indirect path return geographic server load balancer 54 may then precisely time ~~queries~~ transit time requests to each of the servers 56a, 56b, 56c so that the ~~queries~~ transit time requests are received at each of the servers 56a, 56b, 56c at exactly the same time. Thus if pinging determines that the response time for the servers 56a, 56b, and 56c are 100ms, 20ms, and 50ms, respectively, then the indirect path return geographic server load balancer 54 will first send a ~~query~~ transit time request to server 56a, then wait 25ms and send a ~~query~~ transit time request to server 56c (the difference in the response times between server 56a and 56c divided by 2 because the ~~query~~ transit time request is only traveling one-way), then wait another 15ms and send a ~~query~~ transit time request to server 56b (difference in the response times between server 56a and 56b is 80ms, divided by 2 gives 40ms from the time the ~~query~~ transit time request to server 56a was sent). This ensures that the ~~query~~ transit time requests arrive at each of the servers 56a, 56b, and 56c at the same time.

Please replace the second paragraph beginning on page 8 and ending on page 9, with the following amended paragraph:

Since the ~~query~~ transit time requests arrive at each of the servers 56a, 56b, 56c at the same time, the DNS responses from 56a, 56b, 56c to the client DNS server 52 may be sent at the same time. This creates what may be called a one-way race condition. In order to determine the server 56a, 56b, 56c with the fastest transit time, the client DNS server 52 need only determine the server from which the first DNS Response packet it receives came. When a server 56a, 56b, or 56c wins the race it is known to the server by the corresponding HTTP Request coming in from an IP address with similarity to the Client-DNS server. The winning of the one-way race

triggers an update to the indirect path geographic server load balancer with the client DNS servers IP address and the transit time for the race to occur. The indirect path geographic server load balancer then adds this to an entry in its content routing table where it keeps the client-DNS server IP address mapped to the *m* fastest sites that responded to it.

Please replace the second paragraph on page 9 with the following amended paragraph:

In order to remedy this concern, only a certain number of servers will be "active" at any one time for each client-DNS server, treated as independent entities. "Active" means that the server is subject to receive a ~~query~~ transit time request from the indirect path return geographic server load balancer 54 for a particular client DNS server. In a specific embodiment of the present invention, 5 servers will be active at any one time for each client DNS server, although there can be overlap between different client DNS servers. The 5 servers will made up of the 3 servers having the lowest response time to the client-DNS server as well as 2 other servers used for testing (which will be described below).

Please replace the first paragraph on page 13 with the following amended paragraph:

At 104, ~~queries~~ transit time requests are sent to each of the content serving sites, timing the ~~queries~~ transit time requests so that they arrive at each of the content serving sites at the same time by using said response time for each of the content serving sites. This is accomplished by ~~querying~~ sending transit time requests to the content serving sites in order of their response times, longest response time first. The ~~query~~ transit time request for a particular content serving site is delayed from the ~~query~~ transit time request for the content serving site with the longest response time for a time equal to one half of the particular content serving site's response time.

Please replace the second paragraph on page 13 with the following amended paragraph:

At 106, data is received from the fixed location as to the transit times of each of the content serving sites. The fixed location would have measured these transit times by tracking when a response from each of the content serving sites was received (the response from each of the content serving sites being generated upon receipt of the ~~query~~ transit time request from the indirect path return geographic server load balancer). The data may be in a form of a record listing the IP address of fixed location, the transit time from a particular content serving site, and the IP address of that particular content serving site. At 108, the data may then be stored in table for later use.

Please replace the second paragraph on page 14 with the following amended paragraph:

At 114, ~~queries~~ transit time requests are sent to each of *n* fastest content serving sites and *m* other content serving sites, timing the ~~queries~~ transit time requests so that they arrive at each of the *n* fastest content serving sites and *m* other content serving sites at the same time by using said response time for each of the *n* fastest content serving sites and *m* other content serving sites. This is accomplished by ~~querying~~ sending transit time requests to the *n* fastest content serving sites and *m* other content serving sites in order of their response times, longest response time first. The ~~query~~ transit time request for a particular *n* fastest content serving sites or *m* other content serving sites is delayed from the ~~query~~ transit time request for the *n* fastest content serving sites or *m* other content serving sites with the longest response time for a time equal to one half of the particular content serving site's response time.

Please replace the first paragraph on page 15 with the following amended paragraph:

At 116, data is received from the fixed location as to the transit times of each of the *n* fastest content serving sites and *m* other content serving sites. The fixed location would have measured these transit times by tracking when a response from each of the *n* fastest content serving sites and *m* other content serving sites was received (the response from the *n* fastest content serving sites and *m* other content serving sites being generated upon receipt of the ~~query~~ transit time request from the indirect path return geographic server load balancer). The data may be in a form of a record listing the IP address of fixed location, the transit time from a particular content serving site, and the IP address of that particular content serving site. At 118, the data may then be stored in table for later use.

Please replace the second paragraph beginning on page 16 and ending on page 17 with the following amended paragraph:

A ~~query~~ transit time request sender 164 coupled to said response timer determiner 162 and to the content serving sites 152a, 152b, 152c, 152d sends ~~queries~~ transit time requests to each of the content serving sites 152a, 152b, 152c, 152d, timing the ~~queries~~ transit time requests so that they arrive at each of the content serving sites at the same time by using said response time for each of the content serving sites. This is accomplished by ordering the content serving sites in order of their response times, longest response time first, using a response timer orderer 166, and then timing the ~~queries~~ transit time requests with a ~~query~~ transit time request timer 168. The ~~query~~ transit time request for a particular content serving site is delayed from the query for the content serving site with the longest response time for a time equal to one half of the particular content serving site's response time.

Please replace the first paragraph on page 17 with the following amended paragraph:

A data receiver 170 coupled to the fixed location 154 receives data from the fixed location 154 as to the transit times of each of the content serving sites. The fixed location 154 would have measured these transit times by tracking when a response from each of the content serving sites 152a, 152b, 152c, 152d was received (the response from each of the content serving sites 152a, 152b, 152c, 152d being generated upon receipt of the ~~query~~ transit time request from the indirect path return geographic server load balancer 150). The data may be in a form of a record listing the IP address of fixed location, the transit time from a particular content serving site, and the IP address of that particular content serving site. A data storer 172 coupled to said data receiver 170 and to a memory 174 may store the data in a table in memory 174 for later use.

Please replace the second paragraph beginning on page 18 and ending on page 19 with the following amended paragraph:

The ~~query~~ transit time request sender 164 sends ~~queries~~ transit time requests to each of *n* fastest content serving sites and *m* other content serving sites, timing the ~~queries~~ transit time requests so that they arrive at each of the *n* fastest content serving sites and *m* other content serving sites at the same time by using said response time for each of the *n* fastest content serving sites and *m* other content serving sites. This is accomplished by querying the *n* fastest content serving sites and *m* other content serving sites in order of their response times, longest response time first, using said response time orderer. The ~~query~~ transit time request for a particular *n* fastest content serving sites or *m* other content serving sites is delayed from the ~~query~~ transit time request for the *n* fastest content serving sites or *m* other content serving sites with the longest response time for a time equal to one half of the particular content serving site's response time.

Please replace the second paragraph on page 19 with the following amended paragraph:

The data receiver 170 receives data from the fixed location 154 as to the transit times of each of the *n* fastest content serving sites and *m* other content serving sites. The fixed location 154 would have measured these transit times by tracking when a response from each of the *n* fastest content serving sites and *m* other content serving sites was received (the response from the *n* fastest content serving sites and *m* other content serving sites being generated upon receipt of the ~~query~~ transit time request from the indirect path return geographic server load balancer). The data may be in a form of a record listing the IP address of fixed location, the transit time from a particular content serving site, and the IP address of that particular content serving site. A data storer 172 coupled to said data receiver 170 and to a memory 174 may store the data in a table in memory 174 for later use.